To appear in ACM TOG 35(4).

CofiFab: Coarse-to-Fine Fabrication of Large 3D Objects

Peng Song[1] Bailin Deng[2] Ziqi Wang[1] Zhichao Dong[1] Wei Li[1] Chi-Wing Fu[3] Ligang Liu[1*]
[1]University of Science and Technology of China [2]University of Hull [3]The Chinese University of Hong Kong

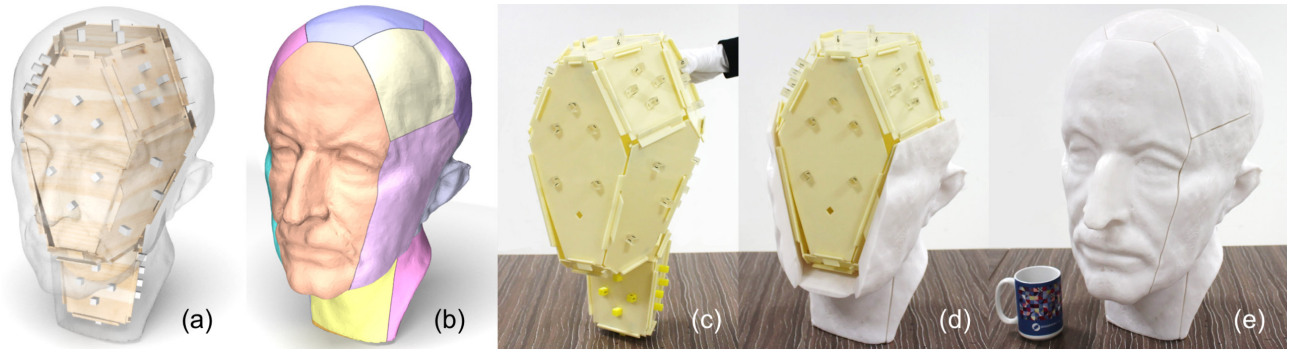


Figure 1: *CofiFab is a novel coarse-to-fine 3D fabrication technique, cost-effectively combining 3D printing and 2D laser cutting for supporting fabrication of large 3D objects. Given the input* MAX PLANCK *model, CofiFab generates a coarse shape proxy with two internal polyhedral bases (a) and an external shell (b) with thin pieces. Each internal base (c), produced by laser cutting, is assembled with a well-designed interlocking joints network. The external shell, realized by 3D printing, is then attached piece by piece to the internal bases with 3D-printed bolts and nuts (d). The final fabricated object is around 24 inches tall (e). A mug is put beside the object as a size reference.*

Abstract

This paper presents CofiFab, a coarse-to-fine 3D fabrication solution, combining 3D printing and 2D laser cutting for cost-effective fabrication of large objects at lower cost and higher speed. Our key approach is to first build coarse internal base structures within the given 3D object using laser cutting, and then attach thin 3D-printed parts, as an external shell, onto the base to recover the fine surface details. CofiFab achieves this with three novel algorithmic components. First, we formulate an optimization model to compute fabricatable polyhedrons of maximized volume, as the geometry of the internal base. Second, we devise a new interlocking scheme to tightly connect the laser-cut parts into a strong internal base, by iteratively building a network of nonorthogonal joints and interlocking parts around polyhedral corners. Lastly, we optimize the partitioning of the external object shell into 3D-printable parts, while saving support material and avoiding overhangs. Besides cost saving, these components also consider aesthetics, stability and balancing. Hence, CofiFab can efficiently produce large objects by assembly. To evaluate CofiFab, we fabricate objects of varying shapes and sizes, and show that CofiFab can significantly outperform previous methods.

Keywords: 3D printing, laser cutting, assembly, interlocking

Concepts: •**Computing methodologies** → *Shape modeling;*

1 Introduction

The recent development in personal fabrication tools, such as milling machines, laser cutters, and 3D printers, offers sufficient shape complexity and resolution to allow users to fabricate various physical objects [Gershenfeld 2007]. *Coarse* fabrication techniques, such as laser cutting, enable quick creation of approximated 3D objects by assembling planar cut plates [McCrae et al. 2011; Schwartzburg and Pauly 2013; Chen and Sass 2015].

3D printing, as a *fine* fabrication technique, is increasingly gaining popularity since it can produce 3D objects of almost any shape with reasonable amount of details. However, it has several limitations. Among them, one critical issue concerns with the size of objects that 3D printing can produce. Typically, this is limited by the printer's working volume, whose dimension (print height) is mostly around 5 to 10 inches for commodity printers.[1] Therefore, to fabricate a larger object, it has to be first partitioned into smaller pieces for printing and then assembled by using connectors [Luo et al. 2012], glue [Vanek et al. 2014; Hu et al. 2014], or interlocking [Song et al. 2015]. Second, 3D printing incurs high material cost and long fabrication time. For a solid object, the cost and time grow cubically (in three dimensions) with the object scale, and such issue remains even if we hollow the object via optimized light-weight interior structures [Stava et al. 2012; Wang et al. 2013; Lu et al. 2014]. Despite existing approaches that address these limitations, it remains challenging to print large objects in a cost-effective way. For example, the 3D model shown in Figure 1(e) (∼24 inches tall) could cost US$150 of material and 191 hours of printing, even if we segment it into pieces and print the pieces with interior structures.

Other than using 3D printing alone to construct parts in object assembly, some very recent works suggested combining 3D printing with other fabrication techniques. For example, we may use LEGO bricks [Mueller et al. 2014] or laser-cut plates [Beyer et al. 2015; Gao et al. 2015] to substitute parts of a 3D-printed object, to reduce the printing time and cost. However, these works mainly focus on the design and rapid prototyping of functional objects, e.g., mouse and soap dispenser; they lack computational support for 3D printing and do not consider important fabrication constraints such as structural integrity, so are unsuitable for fabricating large objects.

In this work, our goal is to develop a computational solution

> *To fabricate large objects by cost-effectively combining 3D printing and 2D laser cutting.*

*Corresponding author: lgliu@ustc.edu.cn (Ligang Liu)

[1]http://www.productchart.com/3d_printers/

Inspired by level-of-detail shape representation, where a 3D shape is decomposed into coarse shape proxies and fine geometric details, we propose a novel *coarse-to-fine* fabrication approach, named *CofiFab*, for fabricating large 3D objects with minimized cost. Specifically, we represent a 3D shape as multiple coarse polyhedral bases inside the object (Figure 1(a)), and a fine geometric shell over the bases (Figure 1(b)). The bases, produced by laser cutters with low material cost and high fabrication speed, are assembled with a well-designed network of interlocking joints (Figure 1(c)). The shell, which can be further decomposed into thin pieces, is realized by 3D printers and attached to the bases with printed bolts and nuts (Figure 1(d,e)).

There are three major challenges in the approach. First, we construct the bases as convex polyhedrons to guarantee the structural integrity [Alexandrov 2005], and maximize the size of these polyhedrons to minimize the fabrication cost. This is an NP-hard problem [Borgefors and Strand 2005; Deits and Tedrake 2015], and little work has been done for 3D cases. In addition, we need to enforce a number of geometric constraints on the polyhedrons for their fabrication and assembly, making the problem even more challenging. Second, we need to tightly connect the planar laser-cut pieces of each polyhedron to ensure the stability of the entire assembly; in particular, we need to further attach 3D-printed pieces on the laser-cut base. We approach this challenge by developing a novel interlocking scheme with nonorthogonal joint models to iteratively assemble and interlock the laser-cut pieces. This method has not been explored in previous works on computing interlocking [Song et al. 2012; Fu et al. 2015] nor works on assembling laser-cut pieces [Schwartzburg and Pauly 2013; Cignoni et al. 2014]. Lastly, we need to carefully plan the layouts of the laser-cut and 3D-printed pieces; here, we have to optimize them simultaneously to tightly couple the pieces together, to reduce the 3D-printing cost, as well as to achieve various desirable properties in the final object, such as aesthetics (avoiding obtrusive cutting seams), symmetry, and balance.

CofiFab enables us to take advantage of the complementary strengths of 3D printing and 2D laser cutting, while satisfying various fabrication requirements and achieving other desirable properties. Altogether, CofiFab has three novel technical contributions:

- First, we formulate an optimization model to compute fabricatable convex polyhedrons of maximized volume inside the target shape. Our formulation not only considers the fabrication requirements and material usage for laser cutting and 3D printing, but also takes into account the aesthetics and partitioning of the object shell and the balance and symmetry of the finished object.
- Second, we develop a novel interlocking scheme with nonorthogonal joint models to construct a stable laser-cut assembly from each computed polyhedron. Our method can iteratively plan a network of joints over the laser-cut parts to immobilize all of them in the assembly, except for a specific key; hence, we can tightly connect the laser-cut parts and assemble a stable laser-cut base.
- Lastly, we optimize not only the layout of the convex polyhedrons but also the layout of the 3D-printed pieces. Here, we consider the partitioning of the object shell in the optimization formulation for convex polyhedrons. This early integration allows us to tightly couple the two fabrication methods, further save 3D-printing material, and avoid obtrusive seams on the final object.

Overall, CofiFab is a new assembly-based fabrication solution, taking the advantages of 3D printing and 2D laser-cutting simultaneously and making large object fabrication practical and cost-effective. Given an input object, CofiFab outputs a set of 3D-printable and laser-cuttable parts with appropriate joints, which are ready for production by 3D printers and laser cutters in parallel. By our optimization method, CofiFab can significantly reduce the overall fabrication cost and time, while considering many important properties, e.g., structural integrity, aesthetics, and balance. By our interlocking scheme, CofiFab can create a stable laser-cut assembly, thus enhancing the structural integrity for fabricating large objects. We verify various advantages of CofiFab and present its improvement in efficiency on fabricating objects of varying shapes and sizes.

2 Related Work

Fabrication by assembling 3D-printed parts. Medellín et al. [2007] employed a regular 3D grid to segment an object into parts and modified the parts to address manufacturing issues: thin section, cusp, and knife edges. Hao et al. [2011] extracted and employed feature lines on objects as guidance in object decomposition. Later, Luo et al. [2012] developed an iterative planar-cut method, aiming to fit decomposed parts in the 3D printing volume while considering factors such as assemblability and aesthetics.

More recently, researchers started to consider additional fabrication issues, e.g., saving printing material, packing parts for printing, and structural strength. Hu et al. [2014] proposed a pyramidal decomposition method to model 3D-printed parts, aiming to reduce support material and printing time. Vanek et al. [2014] improved the 3D printing efficiency by iteratively merging and packing small object parts tightly in a small volume for printing. Alemanno et al. [2014] built cultural heritage replicas by decomposing the object shell into parts connected with box joints. Later, Chen et al. [2015] devised a solid- rather than shell-based decomposition method to generate and tightly pack parts for printing, while Yao et al. [2015] developed a level-set method that iteratively optimizes the object decomposition based on assorted metrics. Song et al. [2015] proposed to strengthen a 3D-printing assembly with mechanical interlocking. Compared to these works, CofiFab constructs an assembly with 3D-printed parts as well as 2D laser-cut parts, aiming to cost-effectively combine them to reduce the overall fabrication cost and time.

Cost-effective 3D printing. Recently, the problem of saving 3D printing material started to draw interests from graphics researchers. Stava et al. [2012] proposed to hollow a 3D-printed object while maintaining its structural strength by adding some internal struts. To improve the structural strength of a printed object, several internal geometric structures were developed: the skin-frame structure by Wang et al. [2013], the honeycomb-like structure by Lu et al. [2014], and the medial axis tree structure by Zhang et al. [2015]. Later, Hornus et al. [2015] constructed the minimal printable enclosure for generating maximized inner cavities for cost-effective 3D printing. In this work, we also hollow 3D-printed parts to save material, but far more than this, CofiFab optimizes combined usage of 3D printing and 2D laser cutting, thus achieving a significant reduction of printing material and so the overall material cost; moreover, it also enhances the structural strength through a strong interlocking laser-cut structure within the fabricated assembly.

3D fabrication using planar pieces. Other than 3D printing, another common way to fabricate a 3D object is by assembling planar pieces. McCrae et al. [2011] created intersecting planar slices that cut through the interior volume of a 3D object and maximize feature coverage based on a user study. Later, Hildebrand et al. [2012] studied the feasibility of constructing planar-slice cardboard assemblies using a binary partitioning data structure, while Schwartzburg and Pauly [2013] further developed an optimization method to consider fabrication, stability and assembly constraints. More recently, Cignoni et al. [2014] proposed to distribute planar slices over a 3D object in a field-aligned manner to better represent the global features that characterize the object appearance. Other than using planar slices that intersect mostly orthogonally, Chen et al. [2013] approximated a 3D shape by a simplified surface mesh, where the planar faces are fabricated and then assembled together.

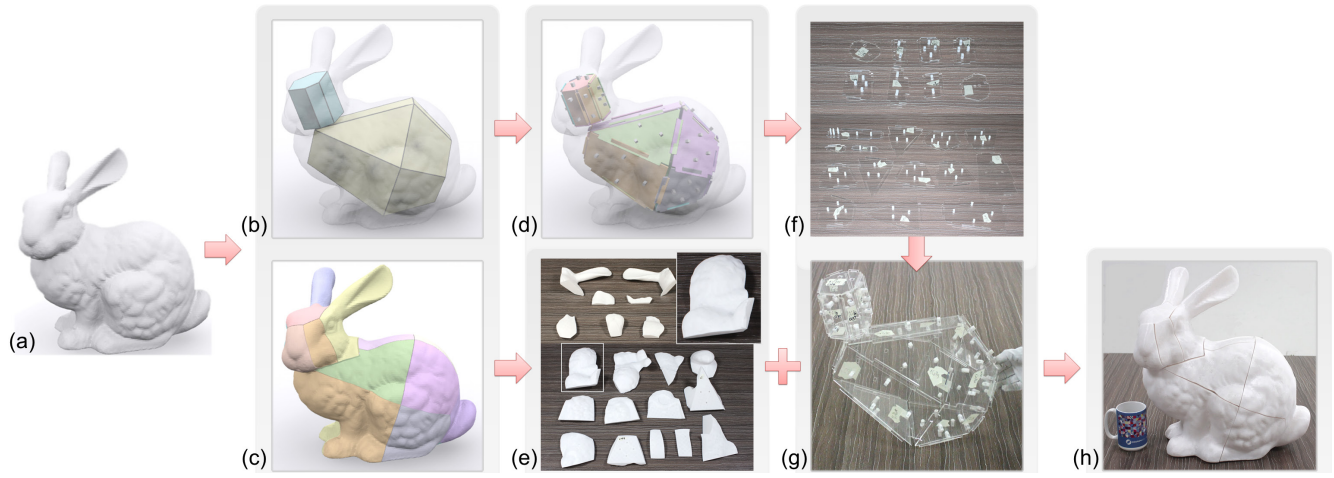


Figure 2: *Overview. (a) input model; (b&c) optimized results: coarse polyhedrons (internal bases) and fine 3D-printed parts (external shell); (d) interlocked laser-cut assemblies; (e) 3D-printed parts; (f) laser-cut parts; (g) assembled laser-cut bases; and (h) final assembled object.*

This work also constructs assembly of planar pieces, but with a different goal. Our assembly need not be a perfect approximation of the input object. Rather, the assembly should help reduce the overall fabrication cost by saving printing material, and at the same time serve as a strong internal base for stably holding the overall assembly. Hence, we formulate a new optimization method to arrange convex polyhedrons for the assembly and a new interlocking scheme to plan a joint network for tightly holding the laser-cut parts together.

Mixed 3D fabrication. Some recent works from the human computer interaction community started to explore the use of other fabrication methods together with 3D printing. Among them, the so-called *low-fidelity fabrication* techniques speed up fabrication by only 3D printing the parts that require high resolution, while realizing the remaining components using faster fabrication methods such as LEGO bricks [Mueller et al. 2014] and laser cutting [Beyer et al. 2015]. However, such approach often leads to lower shape quality for the non-3D-printed parts; in contrast, CofiFab realizes the whole exterior shell by 3D printing, thus ensuring high-quality results, while saving cost and time. Moreover, CofiFab systematically optimizes the part layout to enable fabrication of large objects, which is not considered in any existing approach.

Very recently, Gao et at. [2015] presented *RevoMaker*, a method that builds functional objects using multi-directional 3D printing on top of a laser-cut cuboid; the size and orientation of the cuboid are optimized to save printing material. Compared to RevoMaker, CofiFab employs general convex polyhedrons as internal bases, thus enabling closer approximation of the target shape and more efficient usage of printing material. Furthermore, the object fabricated by RevoMaker is limited in size by the printer's working volume, while CofiFab allows for fabrication of much larger objects. Lastly, RevoMaker requires the modification of 3D printers, whereas CofiFab can directly work with conventional 3D printers and laser cutters.

3 Overview

CofiFab seeks to construct object assembly composed of 3D-printed parts and 2D laser-cut parts with the following objectives:

- *Fabricability.* The 2D laser-cut parts with the joints should be producible by a laser cutter, while each 3D-printed part should fit inside the working volume of a 3D printer.
- *Assemblability.* We should be able to assemble the 2D laser-cut parts into internal bases without blocking, and then assemble the 3D-printed parts onto the bases to produce a finished object.
- *Cost-effectiveness.* We should aim to minimize the overall fabrication cost and time. Since 2D laser cutting is substantially cheaper and faster than 3D printing, we should maximize its usage to reduce 3D printing material in the fabrication; in addition, we should avoid unnecessary 3D-printed parts in the assembly and reduce support material needed in the printing process.
- *Stability.* We should tightly connect the 3D-printed and laser-cut parts together in the finished assembly. Moreover, since we use two different fabrication material and the assembly contains a large empty space inside, we should balance the weight of the parts to ensure that the finished assembly can stand on its own.
- *Aesthetics.* We should consider the object symmetry and avoid obtrusive cutting seams that go across salient object features.
- *Large Object.* Last but not least, we should be able to fabricate large objects, say in the range of 0.5 to 1 meters, which is far greater than the working volume of most existing 3D printers.

Deriving a solution that meets all these objectives is challenging. In particular, the 3D-printed and laser-cut parts have influence on each other; they need to be considered together. CofiFab is a coarse-to-fine fabrication approach, simultaneously optimizing both parts and constructing strong interlocking bases for holding them together, see Figure 2 for the major steps in CofiFab:

1. First, we compute convex polyhedrons to approximate the object interior. In detail, we segment the object into major components, and formulate an optimization to arrange polyhedrons that best approximate these components (Figure 2(b)). Our optimization also considers the partition of the external shell into 3D-printed parts (Figure 2(c)) and incorporates various objectives above.
2. Second, we construct an interlocking laser-cut assembly (Figure 2(d,f,g)) to realize each optimized polyhedron. This is done by a novel corner-based interlocking scheme, two nonorthogonal joint models for connecting the laser-cut parts, and an iterative method that constructs a network of joints to tightly interlock the laser-cut parts into a stable assembly.
3. Lastly, we fabricate the 2D laser-cut and 3D-printed parts using a laser cutter and a 3D printer in parallel (Figure 2(e,f)), assemble the internal laser-cut bases (Figure 2(g)), and attach the 3D-printed parts onto the bases with printed bolts and nuts to assemble the final object (Figure 2(h)). Since the polyhedron optimization considers the 3D-printed parts, we can print them with reduced support material and avoid obtrusive seams.

4 Shape Approximation using Polyhedrons

Shape approximation in CofiFab can be formulated as:

> *Given a 3D object, find a few convex polyhedron(s) of maximized volume to approximate the object interior for reducing the fabrication cost, while considering the various objectives in fabrication and assembled result.*

This problem is non-trivial since we have no prior knowledge on the polyhedrons, yet have to solve the following sub-problems: i) how many polyhedrons to use, ii) where to put them, and iii) what is the topology and geometry of each polyhedron.

Our problem differs from existing works on shape approximation [Cohen-Steiner et al. 2004] [Chen et al. 2013] and convex decomposition [Lien and Amato 2008], since we require the convex polyhedrons to stay strictly inside the object. Our problem is close to the potato peeling [Goodman 1981], which finds a convex polygon/polyhedron of maximum size in a given simple 2D/3D shape. Only a few works have addressed this NP-hard problem [Borgefors and Strand 2005] [Deits and Tedrake 2015], mostly in 2D [Chang and Yap 1986] [Cabello et al. 2014]. In sharp contrast, we employ multiple polyhedrons to approximate the object interior and enforce a number of geometric constraints for fabrication and assembly.

We approach this challenging and unique problem in two steps. First, we identify large quasi-convex components in the object for locating the polyhedrons (sub-problems i and ii). Second, we formulate an optimization to compute the topology and geometry of polyhedrons while considering the various objectives (sub-problem iii).

4.1 Identify Large Quasi-convex Components

To identify large quasi-convex components, we first compute a signed distance function from the object boundary, with positive values in the interior. We then locate local maximum points and progressively grow an internal volume from each of them (see the green volume in Figure 3(a)) by including surrounding points with distance values larger than a threshold $\delta > 0$. Afterwards, we clip off regions that are far away from the computed internal volumes (see the grey regions in Figure 3(b)). Next, we construct a skeleton for each connected component of the remaining shape, and locate points with local minimum signed distance value on the skeletons. These points suggest locations of clipping planes for further segmenting out the quasi-convex components (see Figure 3(c)). By default, a clipping plane is automatically placed orthogonal to the skeleton structure at each detected point. Users can also interactively adjust a clipping plane to improve the result, e.g., when the skeleton does not perfectly align with the object features.

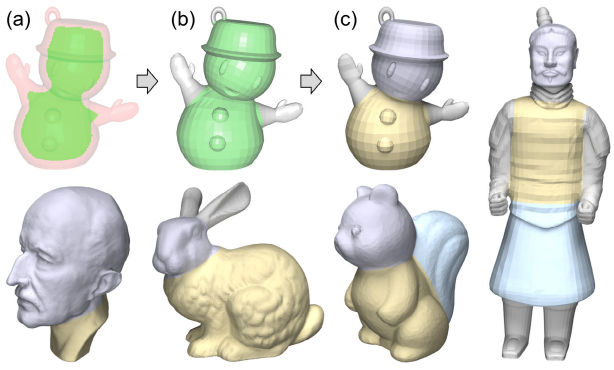


Figure 3: *(a)-(c): identifying quasi-convex components in* SNOWMAN. *More results: quasi-convex components in* MAX PLANCK, BUNNY, SQUIRREL, *and* TERRA COTTA WARRIOR *(left to right).*

4.2 Object Approximation with a Single Polyhedron

For an object with one quasi-convex component, we approximate its interior volume by a single convex polyhedron; we will discuss the case of multiple quasi-convex components later. In general, the more faces the polyhedron has, the better it approximates. However, having excessive faces will complicate the assembly process and increase the fabrication time. Hence, we generate polyhedrons with at most N faces, where N is a user input parameter.

Conceptually, we formulate a constrained optimization problem as

$$\max_{P} \; V(P), \qquad \text{s.t.} \;\; P \in \mathcal{S}_i, \;\; \forall\, i. \tag{1}$$

where P is the solution polyhedron, $V(P)$ its volume, and $\{\mathcal{S}_i\}$ a set of geometric constraints related to fabrication, assembly, physical properties, and aesthetics of the finished model. In general, this problem is non-convex. It requires us to consider the topology and vertex positions of P subject to a large number of constraints. In this work, we take an iterative approach to compute P. We start with a simple convex polyhedron (typically a cuboid), and optimize its vertex positions with fixed connectivity to obtain an initial polyhedron P_0. Afterwards, we iteratively perform the following steps to obtain the next polyhedron P_{i+1} (with more faces) from P_i:

1. Determine a set of candidate topologies for P_{i+1};
2. For each candidate topology, optimize the vertex positions to obtain a candidate shape; and
3. Pick the candidate shape with the largest volume as P_{i+1}.

The process terminates when P_{i+1} has N faces. A key component in this algorithm is the optimization of a polyhedron shape with a fixed topology. More precisely, given the connectivity of the n vertices of a polyhedron, we determine the vertex positions $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_n] \in \mathbb{R}^{3\times n}$ by solving an optimization problem:

$$\min_{\mathbf{P}} \; -V(\mathbf{P}), \quad \text{s.t.} \;\; U_j \le G_j(\mathbf{P}) \le L_j \;\; (j = 1, \ldots, m), \tag{2}$$

where $V(\mathbf{P})$ is the polyhedron volume, and functions $\{G_j\}$ enforce geometric constraints for the polyhedron. The formula for $V(\mathbf{P})$ is given in the supplementary material. In the following, we explain in detail the formulation of the constraint functions, as well as how to determine the candidate topologies.

Convex polyhedron. For each face f_j of a convex polyhedron, its incident vertices should lie on a common plane (f_j), while all other polyhedron vertices should lie on the same side of f_j. We introduce two sets of auxiliary variables: $\mathbf{n}_j \in \mathbb{R}^3$ is the outward unit normal vector of f_j (with condition $\mathbf{n}_j \cdot \mathbf{n}_j = 1$), and $d_j \in \mathbb{R}$ is the signed distance from the plane of f_j to the origin. Then the convex polyhedron conditions lead to constraints for each incident vertex $\mathbf{p}_i$ and each non-incident vertex $\mathbf{p}_k$ of f_j:

$$\mathbf{n}_j \cdot \mathbf{p}_i + d_j = 0, \qquad \mathbf{n}_j \cdot \mathbf{p}_k + d_j \le 0. \tag{3}$$

Assembly requirements. To realize the polyhedron as a laser-cut assembly, we avoid some problematic forms: i) adjacent faces with small dihedral angles; ii) sharp corners on polyhedron faces; and iii) very short edges, since these configurations make it difficult to construct joints between laser-cut parts. For the first two configurations, we enforce feasible ranges, $[\alpha_{\min}, \alpha_{\max}]$ and $[\beta_{\min}, \beta_{\max}]$, for the dihedral angles and the face corner angles, respectively. Thus

$$-\cos\alpha_{\min} \le \mathbf{n}_i \cdot \mathbf{n}_j \le -\cos\alpha_{\max} \tag{4}$$

for each pair of adjacent faces f_i, f_j on the polyhedron, and

$$\cos\beta_{\max} \le \frac{\mathbf{p}_i - \mathbf{p}_j}{\|\mathbf{p}_i - \mathbf{p}_j\|} \cdot \frac{\mathbf{p}_k - \mathbf{p}_j}{\|\mathbf{p}_k - \mathbf{p}_j\|} \le \cos\beta_{\min} \tag{5}$$

for each interior angle (around consecutive vertices $\mathbf{p}_i, \mathbf{p}_j, \mathbf{p}_k$) in each face. To prevent short edges, for each pair of adjacent vertices $\mathbf{p}_i, \mathbf{p}_j$, we require

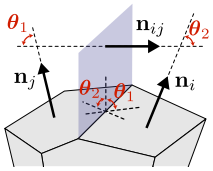
$$\|\mathbf{p}_i - \mathbf{p}_j\| \ge l_{\min}\,, \tag{6}$$

where $l_{\min} > 0$ is a user-specified parameter.

Fabrication requirements. Altogether, we have four requirements for fabrication. First, we require the polyhedron to lie inside the target shape. In particular, to ensure a 3D-printable external shell, the distance from any point on the polyhedron to the target shape surface should be no smaller than a threshold $d_{\min}$. We densely sample the polyhedron surface, and enforce a constraint

$$D_M(\mathbf{q}_i) \ge d_{\min} \tag{7}$$

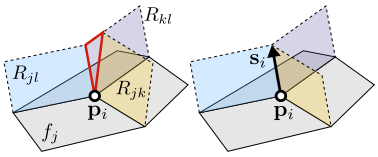
for each sample point $\mathbf{q}_i$, where D_M is the signed distance function to the shape boundary. To relate this constraint to the vertex positions, each sample point is represented as a convex combination of the vertices for its associated polyhedron element (face/edge/vertex). Details on the sampling are provided in the supplementary material.



Second, the layout of the polyhedral edges affects the partition of the external shell into 3D-printed parts. More precisely, for an edge shared by faces f_i, f_j, we create an incident plane R_{ij} with an auxiliary variable $\mathbf{n}_{ij}$ as its normal and conditions: $\mathbf{n}_{ij} \cdot \mathbf{n}_{ij} = 1$ and $\mathbf{n}_{ij}$ is orthogonal to the incident polyhedron edge for R_{ij}. The planes $\{R_{ij}\}$ separate the shell into smaller parts. Each part has a flat base over the polyhedral face for it to align with the print bed (Figure 4). We require the dihedral angle between a partition plane and a polyhedron face to be at least $\pi/2$, to guarantee each 3D-printed part can be attached along the face normal direction without blocking. On the other hand, if the dihedral angle is too large, we will produce thin and large overhangs in the 3D-printed parts, which require additional support material and fabrication time. To avoid this, we enforce the following conditions (see inset above):

$$0 \le \mathbf{n}_{ij} \cdot \mathbf{n}_i \le \cos\gamma \quad \text{and} \quad -\cos\gamma \le \mathbf{n}_{ij} \cdot \mathbf{n}_j \le 0\,. \tag{8}$$

Third, for each vertex $\mathbf{p}_i$, we require its incident partition planes to meet at a common line. Otherwise, they may lead to sharp features that are difficult to fabricate and complicate the assembly structure (see inset, shown in red). Thus we introduce auxiliary variables $\mathbf{s}_i \in \mathbb{R}^3$ for the common unit line direction at each vertex $\mathbf{p}_i$, and enforce an orthogonality constraint between $\mathbf{s}_i$ and the normal of each incident partition plane: R_{jl}, R_{jk}, and R_{kl}.



Fourth, the size of a laser-cut plate is limited by the work area of the cutter. Thus we require each polyhedron face to be bounded by a rectangle whose width w and height h are determined by the work area. For each face f_j, we introduce auxiliary variables $\mathbf{r}_j \in \mathbb{R}^3$ for the rectangle corner, and $\mathbf{e}_{j,1}, \mathbf{e}_{j,2} \in \mathbb{R}^3$ for the unit rectangle edge directions, subject to the condition that $\mathbf{r}_j$ is incident with face f_j, and $\mathbf{n}_j, \mathbf{e}_{j,1}, \mathbf{e}_{j,2}$ are orthogonal to one another. Then the bounding rectangle condition can be enforced for each vertex $\mathbf{p}_i$ on f_j as

$$0 \le (\mathbf{p}_i - \mathbf{r}_j) \cdot \mathbf{e}_{j,1} \le w, \quad 0 \le (\mathbf{p}_i - \mathbf{r}_j) \cdot \mathbf{e}_{j,2} \le h. \tag{9}$$

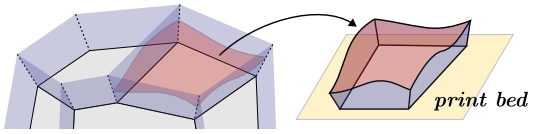
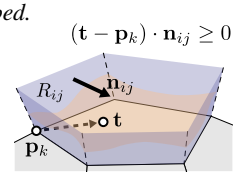


Figure 4: *Using partition planes (purple) associated with polyhedron edges, we can produce for each polyhedron face a 3D-printed part whose base can align with the print bed.*



Aesthetics. Partitioning the shell into smaller parts introduces seams on the final surface. For aesthetic purposes, we allow the user to specify a set of salient regions on the target surface. Each region is assigned to one polyhedron face f_i, with a constraint that the whole region is contained in the 3D-printed part associated with f_i, so that no seams cut through the region. Geometrically, this requires each point $\mathbf{t}$ of the salient region to be enclosed by the partition planes incident with f_i. Let R_{ij} be an incident partition plane of f_i, and $\mathbf{p}_k$ be a vertex incident with R_{ij}. Then the enclosure condition means the signed distance $(\mathbf{t} - \mathbf{p}_k) \cdot \mathbf{n}_{ij}$ from $\mathbf{t}$ to R_{ij} should be non-negative if $\mathbf{n}_{ij}$ points inwards (see inset), or non-positive if $\mathbf{n}_{ij}$ points outwards. To reduce the number of constraints, we only enforce this condition on a subset of points for the salient region. More precisely, we project the region onto its least squares fitting plane, and compute the 2D convex hull of the projection points. Only those points corresponding to the convex hull vertices are subject to the enclosure constraint. When there are more salient regions than polyhedron faces, the user can prioritize the salient regions, and enforce constraints on them based on the priority.

Balance. To ensure the final assembly can stand on its own, we require that its centroid projects along the gravity direction onto a point within the support polygon, i.e., the 2D convex hull of the points on the ground [Prévost et al. 2013]. To compute the centroid, we consider the final model as a combination of hollow polyhedrons made from uniform thin-sheet materials, and a 3D volumetric shell of uniform density. Please see the supplementary material for details.

Implementation. We solve the optimization problem (Eq. (2)) using the interior point solver from the IPOPT library [Wächter and Biegler 2006]. The signed distance function D_M (see Eq. (7)) is constructed by interpolating distance values on a dense grid using the Akima scheme. The gradients of the target function and the constraint functions are evaluated analytically. Since we focus on polyhedrons with a small number of faces, a large portion of constraints are about the signed distance function D_M at the sample points. To accelerate the optimization, we use a small number of sample points when optimizing the candidate shapes for P_{i+1}. After the best candidate is selected, it is further improved by performing the optimization again with more sample points. In general, the solver may fail for some polyhedron candidates. A typical case is that IPOPT does not converge within the maximum number of iterations, when the initial shape is far away from a local solution. However, the best candidate polyhedron often requires only a small number of iterations of the solver, typically less than 500. Thus we set the maximum number of iterations for IPOPT to 1,500, and exclude the candidates for which the solver fails to converge. In our experiments, the solver always succeeds for the majority of candidates, and the incremental optimization always finds a solution.

Polyhedron topology. From an optimized polyhedron P_i, we determine a set of new polyhedrons with more faces, as the candidates for P_{i+1}. Each new polyhedron is computed by a planar cutting of a vertex or an edge, so always adding one new face to the polyhedron. We average the adjacent face normals of the cut vertex/edge as the cutting plane normal. Then the position of the plane is determined

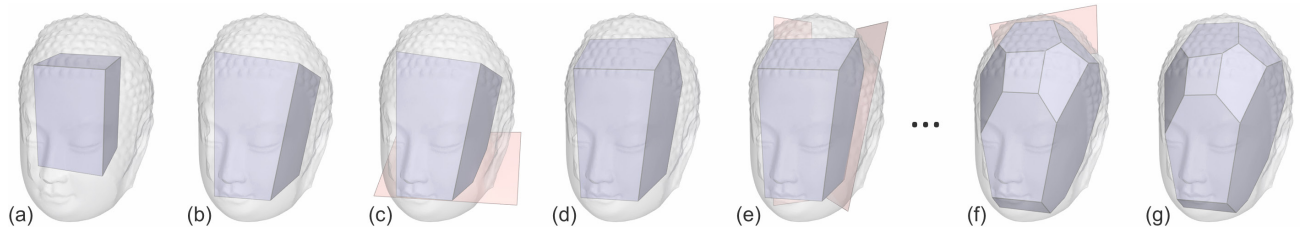


Figure 5: *We iteratively optimize the geometry and topology of a polyhedron to approximate the* BUDDHA HEAD *model. Planar cuts are shown in red and our optimization also supports symmetric cuts, see (e). The volume ratio between the polyhedron and the model is maximized progressively from (a) to (g):* 20.7%, 38.0%, 36.6%, 43.8%, 41.7%, 69.1%, *and* 69.3% *respectively.*

by minimizing the distance from the polyhedron center to the plane, such that for each adjacent edge of the cut vertex/edge, its length in the new polyhedron is no less than half of its original length in P_i. For target shapes with (approximate) mirror symmetry, we can incorporate the symmetry prior into the plane cutting process. Specifically, for a given polyhedron topology, we define its mirror symmetry using a set of symmetric vertex pairs $\{(v_i, v_j)\}$, where $v_i \neq v_j$. A pair of edges $\overline{v_i v_k}$ and $\overline{v_j v_l}$ are defined as symmetric if (v_i, v_j) and (v_k, v_l) are both symmetric pairs, or if (v_i, v_j) is a symmetric pair and $v_k = v_l$. For each vertex/edge that belongs to a symmetric pair, we augment its cutting operation with the cutting of its paired vertex/edge. Using such symmetry-aware cutting, we can obtain polyhedrons that better respect the symmetry of the target shapes, without the need to explicitly enforce symmetry constraints in the optimization. See Figure 5 for an example.

4.3 Object Approximation with Multiple Polyhedrons

In case of multiple quasi-convex components, we construct one polyhedron for each component, see Figure 6. Our optimization algorithm for a single polyhedron can be extended to optimize multiple polyhedrons simultaneously. In addition to the constraints mentioned previously, we need to ensure that the laser-cut bases to be realized from the polyhedrons can be connected using 3D-printed rods to create a connected internal structure for the target shape.

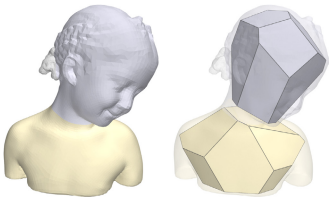
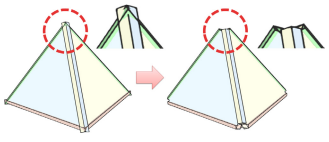


Figure 6: BIMBA *approximated with two polyhedrons.*

In detail, between each pair of adjacent polyhedrons, a pair of faces are selected for the connection. These faces must satisfy the following conditions: 1) they are parallel, with their outward normals pointing towards each other; 2) there exists a cylinder with a large enough radius, whose axis is perpendicular to both faces, and whose ends lie within each face's interior correspondingly; such cylinder should also lie inside the target shape. These conditions ensure multiple rods can be placed to connect the laser-cut bases without going out of the target shape, see supplementary material for details.

5 Interlocking Laser-Cut Assembly



Next, we construct an interlocking laser-cut assembly by creating one laser-cut part for each face in the optimized polyhedron and planning joints on the parts to connect them into a stable assembly structure. We start by thickening each polyhedron face inward by the thickness (τ) of the laser-cut planes and removing intersecting portions near corners.

To plan joints on the parts, we develop a new interlocking scheme and an iterative construction method. Our method can achieve *global interlocking* of laser-cut parts, such that all the laser-cut parts that make up a polyhedron can be immobilized in the finished assembly, except for a special *key* part. Hence, we can tightly interlock all the parts and enhance the structural integrity of the assembly, particularly for supporting large object fabrication.

Joint models. In this work, we employ the mortise-and-tenon and halved (or cross lap) joint models [Craftsmansapce 2015] to connect neighboring laser-cut parts, see Figure 7. We use these two models because they can be realized by laser cutting and they can constrain the connected parts to separate only along certain direction(s). Ideally, if the slit size (s) of the joints exactly matches τ, the laser-cut parts will be connected orthogonally, see Figure 7(a&c). To allow nonorthogonal connections, we can enlarge the slits based on the dihedral angle α between the parts: $s = \tau \frac{|cos\alpha|+1}{sin\alpha}$, see the red arrows in Figure 7(b&d). Note that for halved (HV) joints, we need to enlarge the slit in both laser-cut parts, while for mortise-and-tenon (MT) joints, we only need to enlarge the slit in the mortise part.

To facilitate the discussion on how these joints constrain parts movement, we denote $\vec{e}$ as an edge vector shared between parts and $\vec{n}$ as the normal of the mortise, see Figure 7(b&d). For a nonorthogonal MT joint, we may rotate the tenon (or mortise) about $\vec{e}$, and then take it out along any direction $\perp\vec{e}$ and around $\vec{n}$ (range: $|\pi - 2\alpha|$), see the green arrows in Figure 7(b)(right). Note that we have to carefully plan the roles of mortise and tenon; otherwise, the set of removal directions will change from a range around one normal to another. For a nonorthogonal HV joint, we may still rotate a part relative to the other about $\vec{e}$ (same range), but different from the MT joint, the removal direction is always $\vec{e}$, see Figure 7(d)(right); here, we may choose between $+\vec{e}$ and $-\vec{e}$ when planning the joint.

Lastly, we consider two joint variants, which modify the joint geometry and make joint construction more flexible, see Figure 9. Note that the HV variant was also used in [Cignoni et al. 2014].

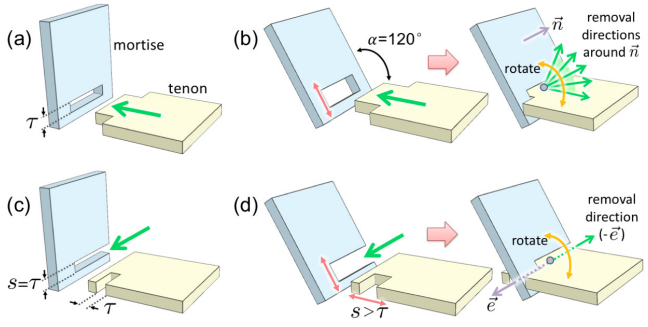


Figure 7: *Joint models: mortise-and-tenon (a&b) and halved (c&d) with orthogonal (a&c) and nonorthogonal connections (b&d). Right column shows possible directions (green) to remove the yellow part.*

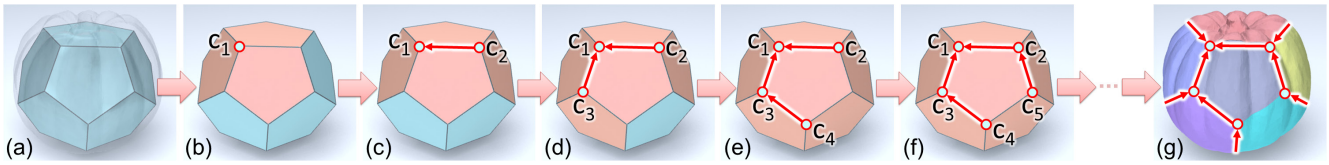


Figure 8: *An illustrative example of the corner walkthrough procedure that iteratively (and globally) interlocks all the parts in the assembly.*

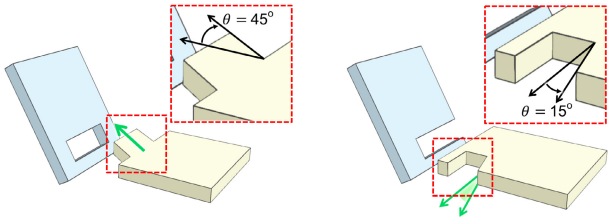


Figure 9: *Left: a mortise-and-tenon variant by shearing the tenon (max: 70°). Right: a halved-joint variant by clipping out an extra portion near the slit of the "yellow" part (max: 20°).*

Corner-based local interlocking. To achieve global interlocking, we construct small local interlocking groups (LIGs) as the building blocks. Here, we propose a *corner-based local interlocking strategy*, inspired by the structure of the laser-cut assembly and an observation that corners of the optimized polyhedron are always incident to three or four faces due to the fabrication constraints in the optimization. Hence, we propose to form corner-based LIG with three or four parts around a corner by planning the joints to immobilize all the laser-cut parts in the LIG, except for a specific part as the *local key*.

By analyzing how the two joint models constrain parts movement and eliminating mirror-reflection and rotational-symmetry cases, we find seven possible LIG configurations that can interlock three laser-cut parts around a corner, see Figure 10. In these cases, only one of the three parts (P_1) is mobile; in the figure, we use two symbols to denote MT and HV joints; the arrows in the symbols indicate part insertion direction, and reveal the role of mortise and tenon in MT joints. After physically trying out these configurations, we found that the two boxed subcases are unstable due to tolerance (P_1 and P_2 in the upper box may be taken out together, similarly for P_1 and P_3 in the lower box), so we ignore these two configurations in the fabrication. Lastly, for LIGs with four laser-cut parts around a corner, we find that there are fifteen usable cases.

Our interlocking scheme. From previous works [Fu et al. 2015; Song et al. 2012], we know that we can impose a dependency between two LIGs by sharing a local key of one of the LIGs; then, the local key of the other LIG can lock all the parts in the combined group. In this work, by considering corner-based LIGs as building blocks, we find that the problem of globally interlocking a laser-cut assembly can be formulated as a *corner walkthrough* problem.

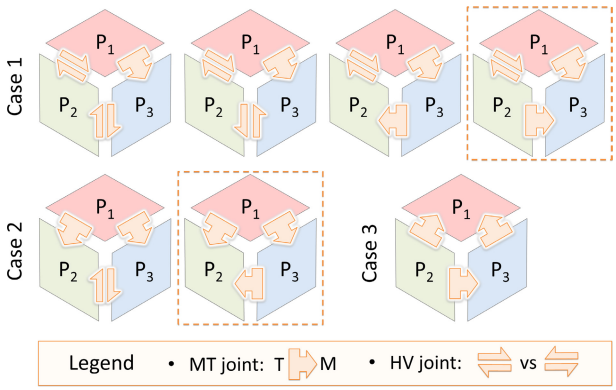


Figure 10: *We find five (excluding the boxed) usable LIG configurations that interlock three laser-cut parts around a corner; P_1 removes along (case 1) the edge shared with P_2, (case 2) a direction on its plane, or (case 3) a direction common to planes of P_2 and P_3.*

That is, after we construct the first LIG around a corner, see c_1 in Figure 8(b), we can pick a neighboring corner (along a polyhedral edge), say c_2, to construct the second LIG, see Figure 8(c). By sharing the local key of the second LIG with the first LIG, we can create a dependency between them, see the arrow in Figure 8(c): the second LIG is locked by the first LIG, so the local key of the first LIG becomes the overall key of the two groups. Then, we can continue to build successive LIGs by picking neighboring corners, see c_3 and c_4 in Figure 8(d&e). In this process, although it is not necessary to cover all the corners (as long as we cover all the parts), we find that we may continue to build more LIGs if it is feasible, and this helps to enhance the interlocking and integrity of the structure, see Figure 8(f). See also Figure 8(g) for the overall dependency group. After this iterative process, each part is locked in some LIGs and each LIG (except the first one) is further locked by some previous LIG(s), so the local key of the first LIG can eventually lock all the parts as the *global key* of the entire laser-cut assembly.

Iterative construction. When using the above scheme to construct interlocking laser-cut assembly, we will have numerous choices for local key, removal direction and corners to pick, as well as various practical issues to address. Hence, we develop the following strategies to guide the iterative construction process:

Global key. First, we prefer a large laser-cut part as the global key. It is because we will later install 3D-printed bolts and nuts on the laser-cut parts for attaching exterior 3D-printed parts, see Figure 11. Hence, having a large key allows our hand to work more easily inside the assembly (before inserting the key) for the installation. Moreover, we prefer a downward insertion direction for the key to avoid slip-off by the gravity, but the user may also pick a specific part as the key or choose a specific insertion direction for it.

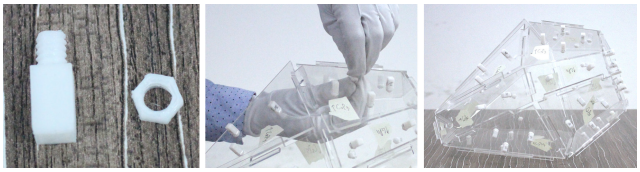


Figure 11: *Install printed bolts and nuts in a laser-cut assembly.*

Removal/insertion direction. The joint variants offer a wider range of choices for removal/insertion directions. This is very important because when a part is removed from an assembly, it may connect with more than one parts in the assembly, except for the last two parts in the disassembly sequence (or equivalently the first two parts in the assembly). Hence, we have to ensure that all the connecting joints around the part permit a common removal direction, so that we can remove it from, or insert it to, the assembly. In this work, we consider the following cases of removal/insertion directions:

Case i) Figure 12 (left): part P is a mortise for all connecting joints, so it can receive a common removal direction (usually its normal) only by moving perpendicular to all surrounding planes.

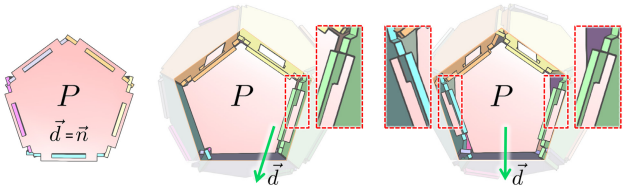


Figure 12: *Planning the joints around a part (P) to remove it along a particular direction: (left) its normal $\vec{n}$, (middle) a shared edge, and (right) average of two shared edges, see the blown-up views for the shared edges; $\vec{d}$ is the resulting removal direction.*

Case ii) Figure 12 (middle): part P moves along a shared edge of a standard HV joint. Since HV joint restricts the part to move only along the shared edge, all other joints must be designed (usually by using joint variants) to permit such removal direction.

Case iii) Figure 12 (right): to allow higher flexibility in joint construction, we may construct a movement direction to be the average of two shared edges, both constructed by using HV joints.

Avoiding no-joint connections. Sometimes we may not be able to create a joint between parts, e.g., P and two of its neighbors in Figure 12 (middle). Due to the removal order, if a part moves toward the shared edge of another part that is to be removed later, we cannot create a joint between them. In particular, a naïve corner walkthrough may produce excessive no-joint connections, which may harm the structural integrity. To avoid this situation, we consider two strategies. First, we prefer to select a corner and local key with fewer adjacent neighbors remained in the assembly; this can reduce the chance for no-joint connections. Second, we try to pick a removal direction that does not result in no-joint connections, particularly by using joint variants, from Figure 12 (left) to Figure 12 (right).

Improving structural integrity. Geometrically, the joint models should perfectly restrict parts movement along a narrow range of directions. However, such a range is always enlarged in practice due to fabrication tolerance, so the joint connections may be loosen, thus making the assembly less steady, see Figure 13 (right). We enhance structural integrity by the following strategies, see result in Figure 13 (left). First, we prefer HV over MT joints since HV joints impose stronger movement restriction. Second, we prefer to produce multiple interlocking, meaning that an LIG is locked by more than one previous LIGs; hence, we continue to construct more LIGs even the dependency graph has covered all the parts in the assembly. Lastly, we estimate the integrity of each part by checking the range of removal directions at its joints; if a non-steady part is found, we will re-select the corner and re-compute the joints.

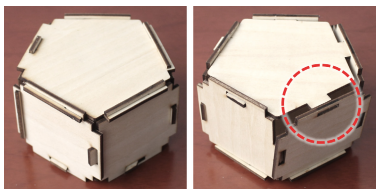


Figure 13: *Our method can generate steady interlocking assembly (left) and avoid non-steady connections due to tolerance (right).*

Overall, the iterative construction process is a greedy approach with backtracking. It iteratively constructs LIGs, local key and joints based on the corner walkthrough scheme. In the end, its output is a set of laser-cuttable shapes with the joints, the assembly/disassembly order, and the removal/insertion direction of each part.

6 Fabricating and Assembling Object Parts

Partition object shell into parts. After generating the interlocking laser-cut frames, we need to partition the object shell into parts

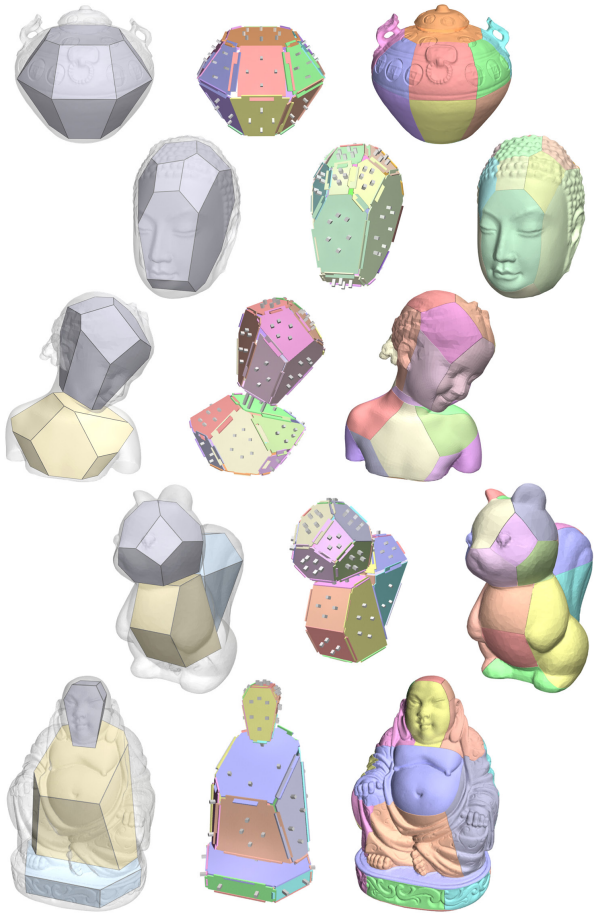
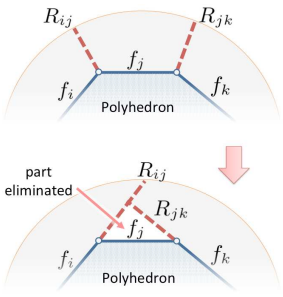


Figure 14: *Results produced by CofiFab, from top to bottom:* CELADON, BUDDHA HEAD, BIMBA, SQUIRREL, *and* BUDDHA.

for 3D printing. For each polyhedron face, we create a corresponding 3D-printed part by clipping the shell using its incident partition planes $\{R_{ij}\}$ produced by the optimization. The clipping planes can also be adjusted to further remove some small 3D-printed parts, which can further save printing material (see inset). Thanks to the fabrication requirements enforced in the optimization, such 3D-printed parts always have flat bases, so they usually do not require additional support material in 3D printing. To connect associated 3D-printed and laser-cut parts, we create holes on them correspondingly, and later connect them using 3D-printed rods and nuts, see Figure 11.



The Final Assembly. First, we insert rods into the holes of each laser-cut part and tighten the rods by using printed nuts. After that, we can assemble these parts to form a laser-cut base. If there are more than one base, we connect them through additional 3D-printed rods and nuts. Finally, we further attach the 3D-printed parts onto the laser-cut bases to reproduce the finished object.

7 Experimental Results

Results. We implement CofiFab in C++ and execute it on a desktop PC with a 3.4GHz CPU and 8GB memory. Figure 14 showcases various object assembly results produced by CofiFab, as well as the associated optimized polyhedron(s) and interlocking laser-cut

	Height (m)	Volume (m^3)			Material Cost (US$) 3D Printing	Material Cost (US$) Laser-cutting	Material Cost (US$) Overall (sum)	Fabrication Time (hours) 3D Printing	Fabrication Time (hours) Laser-cutting	Fabrication Time (hours) Overall (max)	Efficiency (Saved) Material	Efficiency (Saved) Time
Max Planck	0.6	0.038	Hollow	CofiFab	88.5	15.5	104.0	115.2	0.3	115.5	30.5%	39.5%
				Baseline	149.6		149.6	191.0		191.0		
			Solid	CofiFab	280.5	15.5	296.0	326.4	0.3	326.7	59.3%	61.9%
				Baseline	727.7		727.7	857.0		857.0		
Bunny	0.5	0.025	Hollow	CofiFab	54.9	12.8	67.7	71.2	0.3	71.5	26.1%	39.9%
				Baseline	91.6		91.6	119.0		119.0		
			Solid	CofiFab	238.5	12.8	251.3	283.8	0.3	284.1	46.0%	48.6%
				Baseline	465.6		465.6	553.0		553.0		
Terra Cotta Warrior	0.8	0.015	Hollow	CofiFab	49.4	6.9	56.3	69.0	0.3	69.3	23.7%	31.4%
				Baseline	73.8		73.8	101.0		101.0		
			Solid	CofiFab	135.8	6.9	142.7	163.2	0.3	163.5	46.3%	48.5%
				Baseline	265.5		265.5	317.4		317.4		
Garden Pig	0.4	0.011	Hollow	CofiFab	30.3	3.7	34.0	42.4	0.2	42.6	20.6%	26.6%
				Baseline	42.8		42.8	58.0		58.0		
			Solid	CofiFab	100.0	3.7	103.7	120.3	0.2	120.5	50.5%	51.8%
				Baseline	209.3		209.3	250.0		250.0		

Table 1: *CofiFab's performance on saving material and time as compared to a baseline method. The combined volume of 3D-printed parts (solid, CofiFab) for each of the four input models are $0.015m^3$, $0.013m^3$, $0.008m^3$, and $0.005m^3$ (from top to bottom).*

	# Poly	# Part	Approx. Rate	Time (hours)
Celadon	1	36	70.9%	0.21
Buddha Head	1	42	69.3%	0.32
Max Plank	2	39	60.0%	0.64
Bunny	2	44	48.2%	0.72
Bimba	2	41	52.7%	1.45
Garden Pig	2	32	54.8%	0.80
Squirrel	3	54	47.7%	2.22
Buddha	3	48	55.1%	1.82
Terra Cotta Warrior	3	52	48.6%	1.85

Table 2: *Statistics of results from CofiFab. The columns (left to right) refer to the number of polyhedrons, total number of parts, approximation rate, and total time taken to compute the parts.*

base(s). To show the laser-cut and 3D-printed parts more clearly, we present the assembly sequence of KITTEN and SNOW MAN in Figure 18. We physically fabricate some of the results to show the connections among parts, as well as the structural integrity of the final object (see Figures 1, 2 and 17). The internal bases with 3mm plastic or 1mm aluminum plates are produced by a 2D laser cutter of working area 1.5m×1.0m, while the exterior shell and connectors (bolts and nuts) are fabricated by a low-cost FDM printer with printing volume 0.5m×0.4m×0.4m using PLA material. The time taken to assemble the fabricated parts for these models ranges from a few minutes to around an hour, depending on the number of parts and fabrication material involved in the object. Thanks to the stable interlocking bases, the assembled objects are steady and can be repeatedly disassembled and reassembled, see the supplementary video. Moreover, CofiFab can maintain object balance, see the HORSE model in Figure 15.

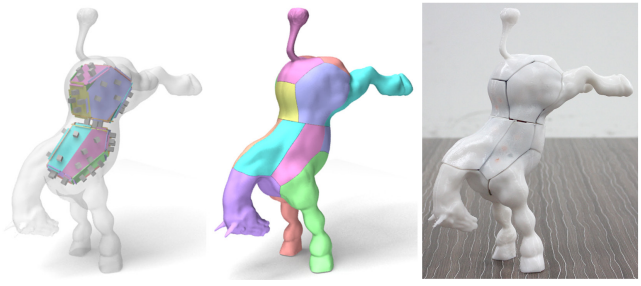


Figure 15: *An assembled* HORSE *model (30cm tall; 18 laser-cut and 16 3D-printed parts) that can balance and stand on two legs.*

Parameters. For the polyhedron optimization algorithm, in all experiments we set $\alpha_{\min} = 35^\circ$, $\alpha_{\max} = 145^\circ$, $\beta_{\min} = 30^\circ$, $\beta_{\max} = 150^\circ$, $l_{\min} = 20$mm, $d_{\min} = 5$mm, and $\gamma = 30^\circ$. The maximum number of polyhedron faces N balances the fabrication cost and manual assembly: a larger N enables closer approximation and reduces the fabrication cost, but also increases the complexity of the assembly process. According to our experience, a value between 20 and 25 usually provides a good trade-off.

Evaluation. We evaluate CofiFab's efficiency in saving material and fabrication time against a baseline that partitions a solid object into 3D printable parts. We perform the comparison in two different ways: i) using solid printed parts for both CofiFab and the baseline; and ii) using hollowed printed parts for both methods, see Table 1. When printing solid parts, the percentages of saved time and cost are close to the volume approximation rates shown in Table 2. This is because printing solid parts requires huge cost and time, where the fabrication cost and time of 2D laser cutting by CofiFab are almost negligible. Even when printing hollowed parts with 15% infill, CofiFab can save around 25% cost and 35% time.

Statistics. Table 2 summarizes the statistics of the results generated by CofiFab: the number of polyhedrons, the number of laser-cut and 3D-printed parts, the approximation rate, and the time taken by the optimization model and the interlocking generation. The approximation rate is computed as the ratio between the total volume of all polyhedrons and the target object volume, which is around or over 50% for all the models in our experiments. The computation time depends on the object shape complexity and the number of polyhedral faces (dominated by the optimization process), while the generation of the interlocking bases usually completes in a few seconds. The overall computation time is significantly lower than the saved fabrication time (see Table 1). In general, CofiFab achieves better results on models with large convex components and few branches. Models with more concave regions or branches usually lead to lower approximation rates. Nevertheless, even for such models, CofiFab achieves approximation rates higher than 40% in our experiments.

Extensions. Laser cutters usually have larger working size than 3D printers. CofiFab can be easily extended to produce more than one 3D-printed parts for each laser-cut part, to take full advantage of the laser cutter and reduce the total number of parts. For example,

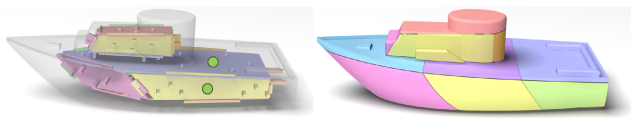


Figure 16: *Multiple 3D-printed parts are attached to the same laser-cut part (marked with a green dot) in this* TOY BOAT *model.*

Figure 17: *Physical fabricated assembly:* HOUSE, GARDEN PIG, *and* TERRA COTTA WARRIOR *models (from top to bottom). The diameters of 3D-printed bolts and nuts are 5mm and 8mm, resp.*

for an elongated object such as TOY BOAT shown in Figure 16, CofiFab can approximate its elongated body by a single polyhedron (the long bottom one), and partition the surrounding object shell into multiple 3D-printed parts to fit into the printing volume.

Lastly, CofiFab can also produce large functional objects. Figure 17 (top) shows a 20cm-tall HOUSE model with a light bulb inside. Since CofiFab produces assemblies based on mechanical interlocking, we can repeatedly disassemble and assemble the fabricated structure and perform maintenance work inside the laser-cut assembly.

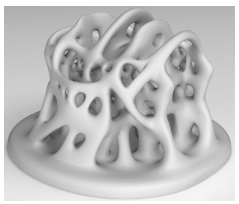
User Interactions. Among the results produced by our method, three of them involve user interaction in their productions: i) BUDDHA (Figure 14): we interactively adjusted the clip plane to make it parallel with the model base to identify the lower quasi-convex component; ii) SNOWMAN (Figure 18): for the lower polyhedron, we specified the key (red) and a horizontal insertion direction; and iii) TOY BOAT (Figure 16): we subdivided four elongated parts.

8 Conclusion

We present a coarse-to-fine approach, called CofiFab, for cost-effective fabrication of large 3D objects. In this approach, we represent an input 3D model by a few coarse (interior) polyhedral bases and a fine (exterior) geometric shell; the interior base is produced by assembling laser-cut slices through a well-designed network of nonorthogonal joints, while the exterior shell is fabricated by using 3D printing in pieces and then attached to the laser-cut bases.

Compared to previous methods, CofiFab is a novel and general computational solution with a number of novel elements: an optimization model that maximizes fabricatable convex polyhedrons for achieving various objectives including cost saving, an interlocking scheme for tightly connecting laser-cut parts into a strong and stable base for holding the assembly structure, and an early integration approach that simultaneously optimizes the layout of 3D-print and laser-cut parts. Powered by these novel technical components, we can achieve various objectives and desired properties, and offer an efficient solution for fabricating large objects. The achievements of CofiFab are evidenced by the real objects we fabricated and the various experiments presented in paper.

Limitations and Future Work. CofiFab relies on a large internal volume for putting the laser-cut bases inside the target shape, thus it fails for models with narrow internal spaces (see inset). Moreover, objects with many concave components cannot be well approximated using a few convex polyhedrons, resulting in less saving in fabrication cost and time; non-convex polyhedrons can provide better approximation for such shapes, but are not guaranteed to be rigid in general. In the future it would be interesting to incorporate non-convex laser-cut bases into CofiFab while ensuring their rigidity. Finally, although the assemblies produced by CofiFab are guaranteed to be rigid and proved to be sufficiently strong in all our experiments, as a future work it would be valuable to perform global structural analysis to further avoid weak spots.



Acknowledgments

We thank the reviewers for the valuable comments. Some of the models are provided by Thingiverse, CGTrader, dayin.la, and SCAN THE WORLD initiative. This work is supported by the National Natural Science Foundation of China (61222206, 11526212, 61403357), the One Hundred Talent Project of the Chinese Academy of Sciences, and the Chinese University of Hong Kong strategic recruitment fund and direct grant (4055061).

References

ALEMANNO, G., CIGNONI, P., PIETRONI, N., PONCHIO, F., AND SCOPIGNO, R. 2014. Interlocking pieces for printing tangible cultural heritage replicas. In *Eurographics Workshop on Graphics and Cultural Heritage*, 145–154.

ALEXANDROV, A. D. 2005. *Convex Polyhedra*. Springer.

BEYER, D., GUREVICH, S., MUELLER, S., CHEN, H.-T., AND BAUDISCH, P. 2015. Platener: Low-fidelity fabrication of 3D objects by substituting 3D print with laser-cut plates. CHI '15, 1799–1806.

BORGEFORS, G., AND STRAND, R. 2005. An approximation of the maximal inscribed convex set of a digital object. In *Image Analysis and Processing - ICIAP 2005*. 438–445.

CABELLO, S., CIBULKA, J., KYNČL, J., SAUMELL, M., AND VALTR, P. 2014. Peeling potatoes near-optimally in near-linear time. SOCG'14, 224:224–224:231.

CHANG, J. S., AND YAP, C. K. 1986. A polynomial solution for the potato-peeling problem. *Discrete & Comp. Geom. 1*, 2, 155–182.

CHEN, L., AND SASS, L. 2015. Fresh Press Modeler: A generative system for physically based low fidelity prototyping. *Computers & Graphics 54*, 157–165.

CHEN, D., SITTHI-AMORN, P., LAN, J. T., AND MATUSIK, W. 2013. Computing and fabricating multiplanar models. *Computer Graphics Forum 32*, 2, 305–315.

CHEN, X., ZHANG, H., LIN, J., HU, R., LU, L., HUANG, Q., BENES, B., COHEN-OR, D., AND CHEN, B. 2015. Dapper: Decompose-and-pack for 3D printing. *ACM Trans. Graph. 34*, 6, 213:1–213:12.

CIGNONI, P., PIETRONI, N., MALOMO, L., AND SCOPIGNO, R. 2014. Field-aligned mesh joinery. *ACM Trans. Graph. 33*, 1, 11:1–11:12.

COHEN-STEINER, D., ALLIEZ, P., AND DESBRUN, M. 2004. Variational shape approximation. *ACM Trans. Graph. 23*, 3, 905–914.

CRAFTSMANSAPCE, 2015. Woodworking joints. http://www.craftsmanspace.com/knowledge/woodworking-joints.html.

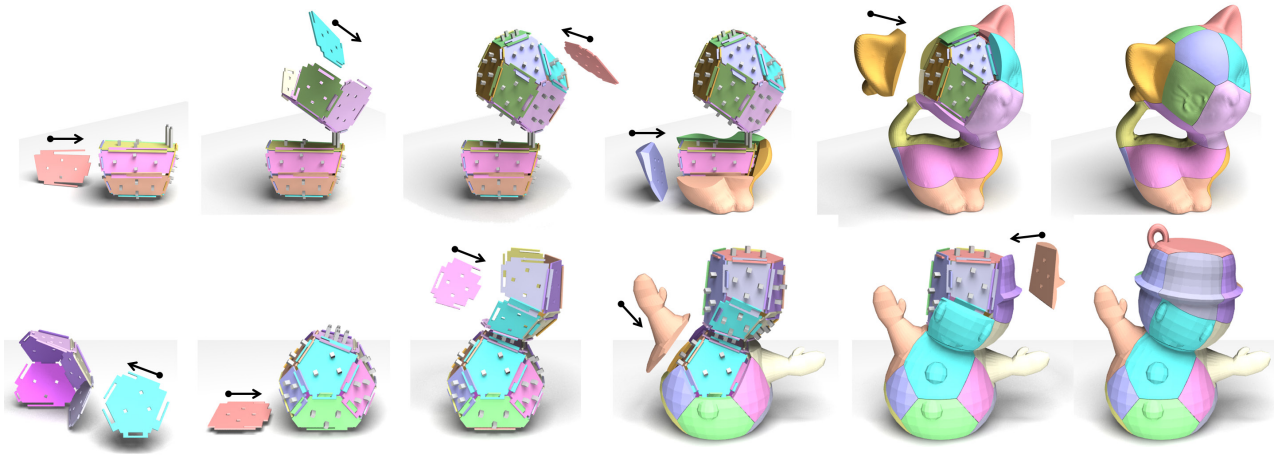


Figure 18: *Snapshots in assembling* KITTEN *(top) and* SNOWMAN *(bottom). The key of each interlocking base is colored in red.*

DEITS, R., AND TEDRAKE, R. 2015. Computing large convex regions of obstacle-free space through semidefinite programming. In *Algorithmic Foundations of Robotics XI*. 109–124.

FU, C.-W., SONG, P., YAN, X., YANG, L. W., JAYARAMAN, P. K., AND COHEN-OR, D. 2015. Computational interlocking furniture assembly. *ACM Trans. Graph. 34*, 4, 91:1–91:11.

GAO, W., ZHANG, Y., NAZZETTA, D. C., RAMANI, K., AND CIPRA, R. J. 2015. RevoMaker: Enabling multi-directional and functionally-embedded 3D printing using a rotational cuboidal platform. UIST '15, 437–446.

GERSHENFELD, N. 2007. *Fab: The Coming Revolution on Your Desktop–from Personal Computers to Personal Fabrication*. Basic Books, Inc., New York, NY, USA.

GOODMAN, J. E. 1981. On the largest convex polygon contained in a non-convex n-gon, or how to peel a potato. *Geometriae Dedicata 11*, 1, 99–106.

HAO, J., FANG, L., AND WILLIAMS, R. E. 2011. An efficient curvature-based partitioning of large-scale STL models. *Rapid Prototyping Journal 17*, 2, 116–127.

HILDEBRAND, K., BICKEL, B., AND ALEXA, M. 2012. crdbrd: Shape fabrication by sliding planar slices. *Computer Graphics Forum 31*, 2, 583–592.

HORNUS, S., LEFEBVRE, S., DUMAS, J., AND CLAUX, F. 2015. Tight printable enclosures for additive manufacturing. Tech. rep., Inria. RR-8712.

HU, R., LI, H., ZHANG, H., AND COHEN-OR, D. 2014. Approximate pyramidal shape decomposition. *ACM Trans. Graph. 33*, 6, 213:1–213:12.

LIEN, J.-M., AND AMATO, N. M. 2008. Approximate convex decomposition of polyhedra and its applications. *Comput. Aided Geom. Des. 25*, 7, 503–522.

LU, L., SHARF, A., ZHAO, H., WEI, Y., FAN, Q., CHEN, X., SAVOYE, Y., TU, C., COHEN-OR, D., AND CHEN, B. 2014. Build-to-last: Strength to weight 3D printed objects. *ACM Trans. Graph. 33*, 4, 97:1–97:10.

LUO, L., BARAN, I., RUSINKIEWICZ, S., AND MATUSIK, W. 2012. Chopper: Partitioning models into 3D-printable parts. *ACM Trans. Graph. 31*, 6, 129:1–129:9.

MCCRAE, J., SINGH, K., AND MITRA, N. J. 2011. Slices: A shape-proxy based on planar sections. *ACM Trans. Graph. 30*, 6, 168:1–168:12.

MEDELLÍN, H., LIM, T., CORNEY, J., RITCHIE, J. M., AND DAVIES, J. B. C. 2007. Automatic subdivision and refinement of large components for rapid prototyping production. *J. Comput. Inf. Sci. Eng. 7*, 3, 249–258.

MUELLER, S., MOHR, T., GUENTHER, K., FROHNHOFEN, J., AND BAUDISCH, P. 2014. faBrickation: Fast 3D printing of functional objects by integrating construction kit building blocks. CHI '14, 3827–3834.

PRÉVOST, R., WHITING, E., LEFEBVRE, S., AND SORKINE-HORNUNG, O. 2013. Make it stand: Balancing shapes for 3D fabrication. *ACM Trans. Graph. 32*, 4, 81:1–81:10.

SCHWARTZBURG, Y., AND PAULY, M. 2013. Fabrication-aware design with intersecting planar pieces. *Computer Graphics Forum 32*, 2, 317–326.

SONG, P., FU, C.-W., AND COHEN-OR, D. 2012. Recursive interlocking puzzles. *ACM Trans. Graph. 31*, 6, 128:1–128:10.

SONG, P., FU, Z., LIU, L., AND FU, C.-W. 2015. Printing 3D objects with interlocking parts. *CAGD 35-36*, 137–148.

STAVA, O., VANEK, J., BENES, B., CARR, N., AND MĚCH, R. 2012. Stress relief: Improving structural strength of 3D printable objects. *ACM Trans. Graph. 31*, 4, 48:1–48:11.

VANEK, J., GALICIA, J. A. G., BENES, B., MĚCH, R., CARR, N., STAVA, O., AND MILLER, G. S. 2014. PackMerger: A 3D print volume optimizer. *Computer Graphics Forum 33*, 6, 322–332.

WÄCHTER, A., AND BIEGLER, L. T. 2006. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program. 106*, 1, 25–57.

WANG, W., WANG, T. Y., YANG, Z., LIU, L., TONG, X., TONG, W., DENG, J., CHEN, F., AND LIU, X. 2013. Cost-effective printing of 3D objects with skin-frame structures. *ACM Trans. Graph. 32*, 6, 177:1–177:10.

YAO, M., CHEN, Z., LUO, L., WANG, R., AND WANG, H. 2015. Level-set-based partitioning and packing optimization of a printable model. *ACM Trans. Graph. 34*, 6, 214:1–214:11.

ZHANG, X., XIA, Y., WANG, J., YANG, Z., TU, C., AND WANG, W. 2015. Medial axis tree-an internal supporting structure for 3D printing. *Comput. Aided Geom. Des. 35-36*, 149–162.